Attorney Docket No. 042626/204671 PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appl. No.: 09/685,131 Confirmation No.: 4846
Applicant(s): Brandon Mitchell Burrell
Filed: October 10, 2000
Art Unit: 2143
Examiner: A. Lezak
Title: SYSTEMS, METHODS, AND COMPUTER PROGRAM PRODUCTS FOR REDIRECTING THE DISPLAY OF INFORMATION FROM A COMPUTER PROGRAM TO A REMOTE DISPLAY TERMINAL

Docket No.: 042626/204671
Customer No.: 00826

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF UNDER 37 CFR § 41.37

This Appeal Brief is filed pursuant to the "Notice of Appeal to the Board of Patent Appeals and Interferences" filed March 21, 2005 and received by the Patent Office on March 24, 2005.

1. *Real Party in Interest.*

The real party in interest in this appeal is American Megatrends, Inc., the assignee of the above-referenced patent application.

2. *Related Appeals and Interferences.*

There are no related appeals and/or interferences involving this application or its subject matter.

3. *Status of Claims.*

Claims 1-75 of the application stand rejected, and these rejections are being appealed. Claims 1, 22, 43, and 64 are independent claims.

4. ***Status of Amendments.***

There are no outstanding amendments to the claims made after the final rejection. The claims on appeal are the same as those filed in the Amendment dated June 11, 2004.

5. ***Summary of Claimed Subject Matter.***

The claimed invention, as embodied in independent Claims 1, 22, 43, and 64, provides systems, methods, and computer program products for redirecting the display of information from a first display to a remote display. (*See* Pat. App., page 1, line 14-16). Display redirection is typically used in network environments to allow a remote user on a remote computer to view the data being displayed on a local computer. (*Id.* at page 4, line 23-26). For example, the remote user may be an IT management person, who is troubleshooting errors on a computer. In these systems, an output redirection handler is used to "grab" display data from the computer and redirect it to the remote user's display. *Id.*

The claimed invention recognizes and solves a problem with many conventional redirection systems. Specifically, Applicant noted that many different computers may run different versions of a piece of software. This required that the remote computer used for redirection had to also contain various versions so as to be compatible with each of the computing systems. *Id.* at page 4, lines 4-14. Further, it was noted that the programs needed to display the information in an appropriate language for the remote user to read. This required that the systems maintain different versions of the software for each language. *Id.* at page 2, line 23-page 3, line 7.

To overcome these issues, the claimed invention provides a unique and non-obvious way to allow output redirection without requiring maintenance of several versions of software. With reference to Figure 2, the claimed invention provides a system containing a display management module 54 that interfaces with the computer program 58 and the display terminal 44. *Id.* at page 18, lines 7-22. The display management module is an intermediate module that interprets display commands from the computer program and controls the display to output the desired information. *Id.* at page 14, lines 25-29. The system further includes data modules 60-66 stored on a computer-readable medium containing text and graphical data used by the computer program 58 to display information on the display terminal 44. Individual 16 bit identification numbers called tokens are associated with the text strings located in the data modules. *Id.* at page 5, lines 2-3.

The source code for the computer program includes the tokens associated with the data strings stored in the language modules, instead of the associated data strings themselves. *Id.* at lines 3-5. In operation, when the computer program wishes to display information on the display terminal, the computer program transmits the token associated with the data string to the display management module. The display management module retrieves the data string associated with the token and also any associated font data. Using the data string and the font data, the display management module displays the information on the display terminal. *Id.* at lines 5-11.

Because the data strings are tokenized and stored in a computer-readable medium separate from the source code of the computer program, these data strings can be easily updated without requiring reprogramming of the source code. *Id.* at lines 12-14. Thus, if a software developer wishes to change the text of the message displayed to the user, or the logo, or the text box displayed to the user, the developer need only change the data string in the appropriate module and not reprogram the source code associated with the computer program. Further, because the font data associated with each character of the data strings is stored in the font and language modules, as opposed to the source code of the computer program, the overall size of the source code of the computer program is reduced. *Id.* at lines 14-21.

6. *Grounds of Rejection to be Reviewed on Appeal.*

Provided below under separate headings is a summary of the rejections to various claims under appeal.

A. Independent Claims 1, 22, 43, and 64

Claims 1-75 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,065,041 to Lum (hereinafter "the '041 Lum patent") in view of U.S. Patent No. 6,141,666 to Tobin (hereinafter "the '666 Tobin patent"). The Office Action dated October 21, 2004 alleges that the '041 Lum patent discloses all aspects of independent Claims 1, 22, 43, and 64 except for the use of tokens to identify text strings. (See Office Action date Oct. 21, 2004, pages 2-3, para. 4. The Office Action argues that the "key-tag sequences" for screen information disclosed in the '041 Lum patent is an obvious equivalent to the recited tokens in the claim. *Id.*

at pages 3-4, para. 5. However, to clarify the argument, the Office Action cites the '666 Tobin patent, which allegedly discloses use of tokens for representing graphical data. *Id.*

B. Dependent Claims 6, 27, 48, and 67

The Office Action dated October 21, 2004 rejects Claims 6, 27, 48, and 67 as obvious light of the combination of the '041 Lum and '666 Tobin patents. The Office Action alleges that the '041 Lum patent discloses a remote handler that stores a current attribute value representing a color of the characters being displayed. *Id.* at pages 5-6, para. 9.

C. Dependent Claims 7, 28, 49, and 68

The Office Action dated October 21, 2004 rejects Claims 7, 28, 49, and 68 as obvious light of the combination of the '041 Lum and '666 Tobin patents. The Office Action alleges that the '041 Lum patent discloses a remote handler that stores a current position of a cursor on the remote display. *Id.* at page 6, para. 10.

D. Dependent Claims 8-17, 29-38, 50-59, and 69-75

The Office Action dated October 21, 2004 rejects Claims 8-17, 29-38, 50-59, and 69-75 as obvious light of the combination of the '041 Lum and '666 Tobin patents. The Office Action alleges that the '041 Lum patent discloses the various aspects of these claims. *Id.* at pages 6-11, paras. 11-23.

7. ***Argument.***

Provide below under separate sub-headings are remarks concerning patentability of various independent and dependent claims.

A. Independent Claims 1, 22, 43, and 64 Are Patentable

As best understood, the '041 Lum patent describes a system that includes a server for providing web pages to remote computers. The server includes a series of web pages that are formatted in different languages or for different types of remote displays. When the web pages are requested by a remote computer, the server first determines what language and/or remote display is used by the remote computer. The server then sends the appropriate web pages in the proper language format and/or display format to the remote computer, such that the contents of the web pages may be properly read by the user at the remote computer.

Applicant respectfully submits that the '041 Lum patent fails to teach or suggest use of tokens to designate text or graphical data stored in data modules for display on the remote display terminal. The '041 Lum patent in effect offers a different solution to that of the claimed invention. Instead of having one version of source code and different data modules that are referenced by tokens in the source code, the system of the '041 Lum patent merely creates completely different web pages for each language or display type. The '041 Lum patent is essentially an example of the type of prior art system that the claimed invention seeks to replace. In the '041 Lum system, the user must maintain various versions of the web pages in each language. Each time the web pages are updated, the user must update each and every version of the web page. This is the exact problem that the claimed invention remedies by placing tokens in the source code and placing text and graphical data corresponding to the tokens in separate modules. With the claimed invention, to add a new language, the user merely need create a new language module. Also, updates to the computer program that do not affect the text or graphics displayed need only be made in the source code and not in multiple versions of the source code.

The Office Action dated October 21, 2004 alleges that the "key-tags" disclosed in the '041 Lum patent correspond to the recited tokens. Applicant respectfully disagrees. As best understood, the key-tags refer to web pages in different languages, not individual text strings stored in modules.

The Office Action also recited the '666 Tobin patent as disclosing a system that uses tokens to represent text and graphics. Applicant is not sure why the '666 Tobin patent is being cited. This reference discloses use of tokens in URL web sites to track where a user has been when surfing the web. If a user starts at a first web site and clicks on a link to a second web site, the system inserts a token into the URL address indicating that the user came and was referred from the first web site. Applicant sees no relevance in this disclosure to a system that uses tokens to represent text and graphics in a computer source code.

In short, neither the '041 Lum patent by itself nor in combination with the '666 Tobin patent teach or suggest providing a computer program having tokens representing text or graphical information to be displayed and individual data modules containing text or graphical data corresponding to the tokens as is recited in independent Claims 1, 22, 43, and 64. Thus, for this reason and the other reasons stated above, Applicant respectfully submits that Claims 1, 22, 43, and 64, as well as the claims that depend therefrom, are patentable over the cited references.

B. Claims 6, 27, 48, and 67 Include Patentable Subject Matter

Claims 6, 27, 48, and 67 recite that the remote display handler stores an attribute representing the color of characters being displayed on the remote display so that this command does not have to be repeatedly sent with each new set of data sent to the remote display. This is nowhere taught or suggested by the '041 Lum patent. The sections of the '041 Lum patent cited in the Office Action as meeting these claims first do not disclose a remote display handler or that the remote handler stores an attribute concerning the color that fonts are being displayed in the remote computer. First, the "scripts" pointed to by the Office Action in the '041 Lum patent are stored on the server, not at the remote computer as is recited in the claims. Second, these scripts are in no way customized for the colors used by each remote display. These scripts are just the scripts used for each web page created by the server.

Finally, because the system of the '041 Lum patent sends whole web pages to the remote computers, it has no need to keep up or track the color used by the remote display, because for each request of data from the remote computer, the server sends the entire web page. Unlike the system of the '041 Lum patent, the system of the claimed invention sends individual data to the remote computers. As such, it is important for it to know with each transmission what color font is already in use for displaying the information on the remote computer. By keeping up with this color attribute at the remote computer, the font color data does not have to be resent with each transmission. This is not of concern the '041 Lum system.

C. Claims 7, 28, 49 and 68 Include Patentable Subject Matter

Claims 7, 28, 49, and 68 serve a function similar to that of Claims 6, 27, 48, and 67. Instead of keeping up with the color used by the remote display, these claims recite a system that keeps up with current cursor position on the remote display, such that this information does not have be sent with each data transmission to the remote display. Hereagain, this aspect is neither taught nor suggested by the '041 Lum patent. As the '041 Lum patent sends entire web pages with each transmission, it does not need to know current cursor position at the remote computer.

D. Claims 8-17, 29-38, 50-59, and 69-75 Include Patentable Subject Matter

Claims 8-17, 29-38, and 50-59 either directly recite or depend from claims that recite individual data modules containing text or graphical data that is associated with tokens that correspond to tokens stored in the computer program. These tokens allow the system to match data to be displayed with the tokens output by the computer program. The '041 Lum patent nowhere teaches or suggests this aspect of the claimed invention. As mentioned, the '041 Lum patent stores entire web pages in the server and sends them to the remote computers. Although these web pages include text and graphical data, the text and graphical data do not have associated tokens. The '041 Lum patent sends the web page file as a whole and does not send individual data in the web page based on tokens received from the computer program.

8. *Claims Appendix.*

Claims currently on appeal are as follows.

1. (Previously Presented) A system for redirecting to a remote display terminal information output by a computer program directed to a first display terminal such that the information can also be displayed at the remote display terminal comprising:

a computer program that contains at least one token stored therein representing one of text data and graphical data to be displayed during operation of said computer program;

a display management module stored on a computer-readable medium in communication with an output of the computer program and the remote display terminal for displaying data output from the computer program directed to the first display terminal to the remote display terminal;

data modules stored on a computer-readable medium containing text data and graphical data representations used by the computer program to display information on a display terminal, wherein at least one of said text data and graphical data representations corresponds to the token stored in said computer program;

an output redirection handler stored on a computer-readable medium in communication with said display management module; and

a data communication link connected between said output redirection handler and the remote display terminal, wherein said display management module receives commands to display data from the computer program, and wherein said display management module, based on the commands from the computer program, provides commands and data from said data modules to said output redirection handler,

wherein said output redirection handler receives the information from said display management module and provides the information to the remote display terminal for display,

wherein to display text or graphical data, said computer program outputs a token representing the text or graphical data to be displayed, and said display management module uses the token to retrieve the text or graphical data from said data modules for display on the remote display terminal.

2. (Original) A system according to Claim 1 further comprising a remote display handler stored on a computer-readable medium in communication with the data communication link and the remote display terminal.

3. (Original) A system according to Claim 2, wherein said display management module provides data and commands based on a predetermined set of commands, and wherein said output redirection handler communicates with said display management module using the predetermined set of commands and provides the commands and data to the remote display handler for display on the remote display terminal.

4. (Original) A system according to Claim 2, wherein said data communication link is a data communication link selected from the group consisting of parallel, serial, and network, and wherein said output redirection handler receives commands and data from said display management module and formats the commands and data for transmission across said data communication link.

5. (Original) A system according to Claim 2, wherein said remote display handler receives commands and data from said output redirection handler and controls the remote display terminal to display the data.

6. (Original) A system according to Claim 2, wherein said remote display handler stores a current attribute value representing a color attribute of the characters being displayed on the

remote display terminal such that subsequent commands to display data on the remote display terminal that do not alter the attribute do not require transmittal of the attribute variable.

7. (Original) A system according to Claim 2, wherein said remote display handler stores a value representing the current position of a cursor on the remote display terminal such that subsequent commands to display data on the remote display terminal do not require data concerning cursor position.

8. (Original) A system according to Claim 1, wherein at least one data module is a language data module including data strings representing language data, wherein each data string is stored in said data modules and designated by a token, and wherein to display a data string, said display management module receives a token associated with the data string from the computer program and a command to display the data string and based on the token accesses said language data module, retrieves the data string associated with the token, and outputs the data string and a command to display the data string to said output redirection handler.

9. (Original) A system according to Claim 1 further comprising:

a plurality of language data modules associated with said display management module, wherein each language data module includes data strings representing language data in a selected language; and

a main language module header stored on a computer-readable medium and associated with said display management module comprising individual pointers indicating the location in a computer-readable storage medium in which each language data module is located, wherein to display a data string in a selected language, said display management module receives a token associated with the data string and a command to display the data string, wherein said display management module accesses said main language module header and retrieves the pointer associated with the language data module corresponding to a preselected desired language for displaying the data string, and wherein said display management module using the pointer, accesses the language data module, retrieves the data string associated with the token, and outputs the data string in the desired language and a command to display the data string to said output redirection handler.

10. (Original) A system according to Claim 8 further comprising a font module stored on a computer-readable medium containing font data for displaying 256 standard and extended ASCII characters.

11. (Original) A system according to Claim 10, wherein at least one data module is a language data module comprising:

a string data area stored on a computer-readable medium that includes data strings representing language data, wherein each character of each data string is a character selected from the group consisting of standard ASCII, extended ASCII, and double byte characters;

an extended ASCII font data area stored on a computer-readable medium for storing font data related to extended ASCII characters that are not displayable using the extended ASCII character font data stored in said font module; and

a double byte character font data area stored on a computer-readable medium for storing font data related to at least one double byte character.

12. (Original) A system according to Claim 11, wherein characters in a data string that are standard ASCII and extended ASCII characters having ASCII codes less than a selected escape code are stored by their ASCII representations in said string data area, while extended ASCII characters and standard ASCII characters, if any, having ASCII codes at least as great as the selected escape code and ASCII characters that identify double byte characters are encoded and the encoded values are stored in said string data area.

13. (Original) A system according to Claim 11, wherein standard ASCII characters and extended ASCII characters having ASCII codes less than the selected escape code are stored by their ASCII code representations in the string data area, and wherein extended ASCII characters having ASCII codes at least as great as the selected escape code and ASCII characters that identify the start of 16 bit double byte characters are encoded into 16 bit values and the encoded values are stored in the string data area.

14. (Original) A system according to Claim 13, wherein double byte characters are sequentially encoded and the 16 bit encoded values representing the double byte characters are stored in said string data area, wherein font data associated with the double byte characters is

stored in the double byte character font data area, and wherein at least one extended ASCII character is encoded in said string data area with an escape code preceding the ASCII representation of the extended ASCII character, and wherein if the extended ASCII character is not displayable with extended ASCII character font data stored in said font module, data for the extended ASCII character is stored in said extended ASCII font data area.

15. (Original) A system according to Claim 14, wherein double byte characters are sequentially encoded such that the first double byte character is represented by a two-byte code having a first byte that is one value greater than the escape code and a second byte equal to zero, and wherein remaining unique double byte characters are encoded with sequential 16 bit code values.

16. (Original) A system according to Claim 15, wherein to display a data string, said display management module receives a token associated with the data string and a command to display the data string from the computer program and accesses the location in the string data area where the data string is located and sequentially outputs the characters of the data string along with a command to display the data string to said output redirection handler.

17. (Original) A system according to Claim 10, wherein when said display management module receives a command to display an ASCII character, said display management module locates the font data associated with the ASCII character in said font module and using the font data outputs the ASCII character to a local display and a command and the character code to said output redirection handler.

18. (Original) A system according to Claim 1 further comprising a logo module stored in a computer-readable medium and associated with said display management module, wherein said logo module contains graphical data for display of at least one logo, and wherein when said display management module receives a command to display the logo, said display management module locates the graphical data associated with the logo in said logo module and outputs the logo image to a local display and a command and data necessary to display a representation of the logo to the output redirection handler.

19. (Original) A system according to Claim 1 wherein said display management module

outputs data for displaying a progress bar and a command to display a representation of the progress bar to said output redirection handler.

20. (Original) A system according to Claim 1, wherein said display management module outputs data for displaying a box defining an area on a display terminal and a command to display the box to said output redirection handler.

21. (Original) A system according to Claim 20, wherein said display management module outputs data for displaying a box defining an area on a display terminal, wherein said display management module further outputs data to be displayed within the defined area of the box, and wherein said display management module outputs commands to scroll the data displayed within the defined area, while any data displayed on other portions of the remote display terminal remain at the same position.

22. (Previously Presented) A method for redirecting to a remote display terminal information output by a computer program directed to a first display terminal such that the information can also be displayed at the remote display terminal comprising the steps of:

providing a computer program that contains at least one token stored therein representing one of text data and graphical data to be displayed during operation of said computer program;

providing on a computer-readable medium data modules containing text and graphical representation data used by the computer program to display information, wherein at least one of said text data and graphical data representations corresponds to the token stored in said computer program;

receiving a command to display data from the computer program;

retrieving data associated with the command from the data modules;

providing a command and data from the data modules for display;

redirecting the command and data to be displayed to the remote display terminal an using an output redirection handler stored on a computer-readable medium; and

displaying the data on the remote display terminal,

wherein to display text or graphical data, said computer program outputs a token representing the text or graphical data to be displayed, and said retrieving step uses the token to retrieve the text or graphical data from said data modules for display on the remote display terminal.

23. (Original) A method according to Claim 22, wherein said providing a command and data step provides data based on a predetermined set of commands, and wherein said redirecting step, using the predetermined set of commands, redirects the data to the remote display terminal.

24. (Original) A method according to Claim 23 further comprising the step of transmitting the command and data from said redirecting step on a data communication link to the remote display terminal.

25. (Original) A method according to Claim 24, wherein the data communication link is a data communication link selected from the group consisting of parallel, serial, and network, and wherein said transmitting step further comprises the step of formatting the commands and data from said providing commands and data step for transmission across the data communication link.

26. (Original) A method according to Claim 24, wherein said displaying step receives commands and data from said transmitting step and controls the remote display terminal to display the data from the computer program.

27. (Original) A method according to Claim 22 further comprising the step of storing a current attribute value representing a color attribute of the characters being displayed on the remote display terminal such that subsequent commands to display data on the remote display terminal from said providing a command and data step that do not alter the attribute do not require output of the attribute variable.

28. (Original) A method according to Claim 22 further comprising the step of storing a value representing the current position of a cursor on the remote display terminal such that subsequent commands to display data on the remote display terminal from said providing a command and data step do not require data concerning cursor position.

29. (Original) A method according to Claim 22, wherein said providing data modules step provides at least one data module that is a language data module including data strings

representing language data, wherein each data string is stored in the language data modules and designated by a token, and wherein to display a data string on the remote terminal display, said receiving step receives a token associated with the data string and a command to display the data string from the computer program and based on the token said retrieving step accesses the data modules and retrieves the data string associated with the token, and wherein said providing a command and data step provides the data string and a command to display the data string.

30. (Original) A method according to Claim 22, wherein said providing data modules step provides a plurality of language data modules on a computer-readable medium, wherein each language data module includes data strings representing language data in a selected language, and wherein said providing data modules step further provides a main language module header on a computer-readable medium comprising individual pointers indicating the location in a computer-readable storage medium in which each language data module is located, wherein to display a data string in a selected language, said receiving step receives a token associated with the data string and a command to display the data string from the computer program, wherein said retrieving step accesses the main language module header and retrieves the pointer associated with the language data module corresponding to a preselected desired language for displaying the data string, wherein said retrieving step using the pointer, accesses the language data module and retrieves the data string associated with the token, and wherein said providing a command and data step provides the data string for display on the remote display terminal.

31. (Original) A method according to Claim 29, wherein said providing step further provides a font module stored on a computer-readable medium containing font data for displaying 256 standard and extended ASCII characters.

32. (Original) A method according to Claim 31, wherein said providing data modules step provides at least one data module that is a language data module, wherein said providing data modules step provides:

a string data area stored on a computer-readable medium that includes data strings representing language data, wherein each character of each data string is a character selected from the group consisting of standard ASCII, extended ASCII, and double byte characters;

an extended ASCII font data area stored on a computer-readable medium for storing font data related to extended ASCII characters that are not displayable using the extended ASCII character font data stored in the font module; and

a double byte character font data area stored on a computer-readable medium for storing font data related to characters that are double byte characters.

33. (Original) A method according to Claim 32, wherein said providing step provides a string data area wherein characters in a data string that are extended ASCII characters and standard ASCII characters, if any, having ASCII codes less than a selected escape code are stored by their ASCII representations in the string data area, while extended ASCII characters and standard ASCII, if any, having ASCII codes at least as great as the selected escape code and ASCII characters that identify the start of 16 bit double byte characters are encoded into 16 bit values and the encoded values are stored in the string data area.

34. (Original) A method according to Claim 32, wherein said providing step provides a string data area wherein characters in a data string that are standard ASCII characters and extended ASCII characters having ASCII codes less than a selected escape code are stored by their ASCII representations in the string data area, while extended ASCII characters having ASCII codes at least as great as the selected escape code and ASCII characters that identify the start of 16 bit double byte characters are encoded into 16 bit values and the encoded values are stored in the string data area.

35. (Original) A method according to Claim 34, wherein said providing step provides a string data area including double byte characters that are sequentially encoded and the encoded values representing the double byte characters are stored in the string data area, wherein font data associated with the double byte characters is stored in the double byte character font data area, and wherein at least one extended ASCII character is encoded as a 16 bit value in the string data area with an escape code preceding the ASCII representation of the extended ASCII character, and wherein if the extended ASCII character is not displayable with the extended ASCII character font data stored in the font module, data for the extended ASCII character is stored in the extended ASCII font data area.

36. (Original) A method according to Claim 35, wherein said providing step provides a string data area including double byte characters that are sequentially encoded such that the first double byte character is represented by a two-byte code having a first byte that is one value greater than the escape code and a second byte equal to zero, and wherein remaining unique double byte characters are encoded with sequential 16 bit code values.

37. (Original) A method according to Claim 35, wherein to display a data string on a remote display terminal, said receiving step receives the token associated with the data string and a command to display the data string from the computer program and said retrieving step accesses the location in the string data area where the data string is located, and wherein said providing a command and data step sequentially outputs the characters of the data string along with a command to display the data string to the output redirection handler.

38. (Original) A method according to Claim 31, wherein to display an ASCII character, said retrieving step locates the font data associated with the ASCII character in the font module, and wherein said providing a command and data step, using the font data, outputs the ASCII character to a local display and a command and the character code to display the character to the output redirection handler.

39. (Original) A method according to Claim 22, wherein said providing data modules step further provides a logo module stored on a computer-readable medium, wherein the logo module contains graphical data for display of at least one logo, wherein when said receiving step receives a command to display the logo, said retrieving step locates the graphical data associated with the logo in the logo module, and wherein said providing a command and data step, using the logo data, provides the logo image to the local display and a command to display a representation of the logo to the output redirection handler.

40. (Original) A method according to Claim 22, wherein said providing a command and data step provides data for displaying a progress bar and a command to display the progress bar representation to the output redirection handler.

41. (Original) A method according to Claim 22, wherein said providing a command and data step provides data for displaying a box defining an area on a display terminal and a

command to display the box to the output redirection handler.

42. (Original) A method according to Claim 41, wherein said providing a command and data step provides data for displaying a box defining an area on a display terminal and data to be displayed within the defined area of the box, and wherein said providing a command and data step provides a command to scroll the data displayed within the defined area, while any data displayed on other portions of the remote display terminal remain at the same position.

43. (Previously Presented) A computer program product for redirecting to a remote display terminal information output by a computer program directed to a first display terminal such that the information can also be displayed at the remote display terminal, wherein the computer program product comprises:

a computer-readable storage medium having computer readable program code means embodied in said medium, said computer-readable program code means comprising:

computer program that contains at least one token stored therein representing one of text data and graphical data to be displayed during operation of said computer program;

first computer instruction means for providing on a computer-readable medium data modules containing text and graphical data used by the computer program to display text information and a representation of graphical images on the remote display terminal, wherein at least one of said text data and graphical data representations corresponds to the token stored in said computer program;

second computer instruction means for receiving a command to display data from the computer program;

third computer instruction means for retrieving data associated with the command from the data modules;

fourth computer instruction means for providing a command and data from the data modules for display;

fifth computer instruction means for redirecting the command and data to be displayed to the remote display terminal using an output redirection handler stored on a computer-readable medium; and sixth computer instruction means for displaying the data on the remote display terminal

wherein to display text or graphical data, said computer program outputs a token representing the text or graphical data to be displayed, and said third computer instruction means uses the token to retrieve the text or graphical data from said data modules for display on the remote display terminal.

44. (Original) A computer program product according to Claim 43, wherein said fourth computer instruction means provides data based on a predetermined set of commands, and wherein said fifth computer instruction means, using the predetermined set of commands, redirects the data for display on the remote display terminal.

45. (Original) A computer program product according to Claim 43 further comprising seventh computer instruction means for transmitting the command and data from said fifth computer instruction means on a data communication link to the remote display terminal.

46. (Original) A computer program product according to Claim 45, wherein the data communication link is a data communication link selected from the group consisting of parallel, serial, and network, and wherein said seventh computer instruction means formats the commands and data from said fifth computer instruction means for transmission across the data communication link.

47. (Original) A computer program product according to Claim 45, wherein said sixth computer instruction means receives commands and data from said seventh computer instruction means and controls the remote display terminal to display the data from the computer program.

48. (Original) A computer program product according to Claim 43 further comprising seventh computer instruction means for storing a current attribute value representing a color attribute of the characters being displayed on the remote display terminal such that subsequent commands to display data on the remote display terminal from said fourth computer instruction means that do not alter the attribute do not require output of the attribute variable.

49. (Original) A computer program product according to Claim 43 further comprising seventh computer instruction means for storing a value representing the current position of a cursor on the remote display terminal such that subsequent commands to display data on the remote display terminal do not require data concerning cursor position.

50. (Original) A computer program product according to Claim 43, wherein said first computer instruction means provides at least one data module that is a language data module including data strings representing language data, wherein each data string is stored in the language data modules and designated by a token, and wherein to display a data string on the remote terminal display, said second computer instruction means receives a token associated with the data string along with a command to display the data string from the computer program and based on the token said third computer instruction means accesses the data modules and retrieves the data string associated with the token, and wherein said fourth computer instruction means provides the data string and a command to display the data string.

51. (Original) A computer program product according to Claim 43, wherein said first computer instruction means provides a plurality of language data modules on a computer-readable medium, wherein each language data module includes data strings representing language data in a selected language, and wherein said first computer instruction means further provides a main language module header on a computer-readable medium comprising individual pointers indicating the location in a computer-readable storage medium in which each language data module is located, wherein to display a data string in a selected language, said second computer instruction means receives a token associated with the data string along with a command to display the data string from the computer program, wherein said third computer instruction means accesses the main language module header and retrieves the pointer associated with the language data module corresponding to a preselected desired language for displaying the data string, wherein said third computer instruction means, using the pointer, accesses the language data module and retrieves the data string associated with the token, and wherein said fourth computer instruction means provides the data string for display on the remote display terminal.

52. (Original) A computer program product according to Claim 42, wherein said first

computer instruction means provides a font module stored on a computer-readable medium containing font data for displaying 256 standard and extended ASCII characters.

53. (Original) A computer program product according to Claim 52, wherein said first computer instruction means provides at least one data module that is a language data module, wherein said first computer instruction means provides:

a string data area stored on a computer-readable medium that includes data strings representing language data, wherein each character of each data string is a character selected from the group consisting of standard ASCII, extended ASCII, and double byte characters;

an extended ASCII font data area stored on a computer-readable medium for storing font data related to extended ASCII characters that are not displayable using the extended ASCII character font data stored in the font module; and

a double byte character font data area stored on a computer-readable medium for storing font data related to characters that are double byte characters.

54. (Original) A computer program product according to Claim 53, wherein said first computer instruction means provides a string data area wherein characters in a data string that are extended ASCII characters and standard ASCII characters, if any, having ASCII codes less than a selected escape code are stored by their ASCII representations in the string data area, while extended ASCII characters and standard ASCII, if any, having ASCII codes at least as great as the selected escape code and ASCII characters that identify the start of 16 bit double byte characters are encoded into 16 bit values and the encoded values are stored in the string data area.

55. (Original) A computer program product according to Claim 53, wherein said first computer instruction means provides a string data area wherein characters in a data string that are standard ASCII characters and extended ASCII characters having ASCII codes less than a selected escape code are stored by their ASCII representations in the string data area, while extended ASCII characters having ASCII codes at least as great as the selected escape code and ASCII characters that identify the start of 16 bit double byte characters are encoded into 16 bit values and the encoded values are stored in the string data area.

56. (Original) A computer program product according to Claim 52, wherein said first computer instruction means provides a string data area including double byte characters that are sequentially encoded and the encoded values representing the double byte characters are stored in the string data area, wherein font data associated with the double byte characters is stored in the double byte character font data area, and wherein at least one extended ASCII character is encoded as a 16 bit value in the string data area with an escape code preceding the ASCII representation of the extended ASCII character, and wherein if the extended ASCII character is not displayable with the extended ASCII character font data stored in the font module, data for the extended ASCII character is stored in the extended ASCII font data area.

57. (Original) A computer program product according to Claim 56, wherein said first computer instruction means provides a string data area including double byte characters that are sequentially encoded such that the first double byte character is represented by a two-byte code having a first byte that is one value greater than the escape code and a second byte equal to zero, and wherein remaining unique double byte characters are encoded with sequential 16 bit code values.

58. (Original) A computer program product according to Claim 56, wherein to display a data string on a remote display terminal, said second computer instruction means receives the token associated with the data string and a command to display the data string from the computer program and said third computer instruction means accesses the location in the string data area where the data string is located, and wherein said fourth computer instruction means sequentially outputs the characters of the data string along with a command to display the data string.

59. (Original) A computer program product according to Claim 52, wherein to display an ASCII character, said third computer instruction means locates the font data associated with the ASCII character in the font module, and wherein said fourth computer instruction means, using the font data, outputs the ASCII character to a local display and a command and the character code to the output redirection handler.

60. (Original) A computer program product according to Claim 43, wherein said first computer instruction means further provides a logo module stored on a computer-readable

medium, wherein the logo module contains graphical data for display of at least one logo, wherein when said second computer instruction means receives a command to display the logo, said third computer instruction means locates the graphical data associated with the logo in the logo module, and wherein said fourth computer instruction means, using the logo data, outputs the logo image to the local display and a command and data necessary to display a representation of the logo to the output redirection handler.

61. (Original) A computer program product according to Claim 43, wherein said fourth computer instruction means provides data for displaying a progress bar and a command to display the progress bar representation.

62. (Original) A computer program product according to Claim 43, wherein said fourth computer instruction means provides data for displaying a box defining an area on a display terminal and a command to display the box.

63. (Original) A computer program product according to Claim 62, wherein said fourth computer instruction means provides data for displaying a box defining an area on a display terminal and data to be displayed within the defined area of the box, and wherein said fourth computer instruction means provides a command to scroll the data displayed within the defined area, while any data displayed on other portions of the remote display terminal remain at the same position.

64. (Previously Presented) A system for redirecting to a remote display terminal information output by a BIOS POST computer program directed to a first display terminal such that the information can also be displayed at the remote display terminal comprising:

a BIOS POST computer program that contains at least one token stored therein representing one of text data and graphical data to be displayed during operation of said computer program;

a display management module stored on a computer-readable medium in communication with an output of the BIOS POST computer program and the remote display terminal for displaying data from the computer program on a display terminal;

data modules stored on a computer-readable medium containing text data and graphical data representations used by the computer program to display information on a display terminal, wherein at least one of said text data and graphical data representations corresponds to the token stored in said BIOS POST computer program;

an output redirection handler stored on a computer-readable medium in communication with said display management module; and

a data communication link connected between said output redirection handler and the remote display terminal,

wherein said display management module receives commands to display data from the computer program, and wherein said display management module, based on the commands from the computer program, provides commands and data from said data modules to said output redirection handler for displaying information on the remote display terminal, wherein to display text or graphical data, said BIOS POST computer program outputs a token representing the text or graphical data to be displayed, and said display management module uses the token to retrieve the text or graphical data from said data modules for display on the remote display terminal.

65. (Previously Presented) A system according to Claim 64 further comprising a remote display handler stored on a computer-readable medium in communication with the data communication link and the remote display terminal.

66. (Previously Presented) A system according to Claim 65, wherein said display management module provides data and commands based on a predetermined set of commands, and wherein said output redirection handler communicates with said display management module using the predetermined set of commands and provides the commands and data to the remote display handler for display on the remote display terminal.

67. (Previously Presented) A system according to Claim 65, wherein said remote display handler stores a current attribute value representing a color attribute of the characters being displayed on the remote display terminal such that subsequent commands to display data on the remote display terminal that do not alter the attribute do not require transmittal of the attribute variable.

68. (Previously Presented) A system according to Claim 65, wherein said remote display handler stores a value representing the current position of a cursor on the remote display terminal such that subsequent commands to display data on the remote display terminal do not require data concerning cursor position.

69. (Previously Presented) A system according to Claim 64, wherein at least one data module is a language data module including data strings representing language data, wherein each data string is stored in said data modules and designated by a token, and wherein to display a data string, said display management module receives a token associated with the data string from the computer program and a command to display the data string and based on the token accesses said language data module, retrieves the data string associated with the token, and outputs the data string and a command to display the data string to said output redirection handler.

70. (Previously Presented) A system according to Claim 64 further comprising:

a plurality of language data modules associated with said display management module, wherein each language data module includes data strings representing language data in a selected language; and

a main language module header stored on a computer-readable medium and associated with said display management module comprising individual pointers indicating the location in a computer-readable storage medium in which each language data module is located, wherein to display a data string in a selected language, said display management module receives a token associated with the data string and a command to display the data string, wherein said display management module accesses said main language module header and retrieves the pointer associated with the language data module corresponding to a preselected desired language for displaying the data string, and wherein said display management module using the pointer, accesses the language data module, retrieves the data string associated with the token, and outputs the data string in the desired language and a command to display the data string to said output redirection handler.

71. (Previously Presented) A system according to Claim 69 further comprising a font module stored on a computer-readable medium containing font data for displaying 256 standard

and extended ASCII characters.

72. (Previously Presented) A system according to Claim 64 further comprising a logo module stored in a computer-readable medium and associated with said display management module, wherein said logo module contains graphical data for display of at least one logo, and wherein when said display management module receives a command to display the logo, said display management module locates the graphical data associated with the logo in said logo module and outputs the logo image to a local display and a command and data necessary to display a representation of the logo to the output redirection handler.

73. (Previously Presented) A system according to Claim 64 wherein said display management module outputs data for displaying a progress bar and a command to display a representation of the progress bar to said output redirection handler.

74. (Previously Presented) A system according to Claim 64, wherein said display management module outputs data for displaying a box defining an area on a display terminal and a command to display the box to said output redirection handler.

75. (Previously Presented) A system according to Claim 74, wherein said display management module outputs data for displaying a box defining an area on a display terminal, wherein said display management module further outputs data to be displayed within the defined area of the box, and wherein said display management module outputs commands to scroll the data displayed within the defined area, while any data displayed on other portions of the remote display terminal remain at the same position.

CONCLUSION

For at least the foregoing reasons, Applicant respectfully requests that the rejections be reversed.

Respectfully submitted,

W. Kevin Ransom
Registration No. 45,031

CUSTOMER NO. 00826
ALSTON & BIRD LLP
Bank of America Plaza
101 South Tryon Street, Suite 4000
Charlotte, NC 28280-4000
Tel Charlotte Office (704) 444-1000
Fax Charlotte Office (704) 444-1111

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to:
Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on May 24, 2005.

Elaine Kelly